VII International Conference on Computational Methods in Marine Engineering
MARINE 2017
M. Visonneau, P. Queutey and D. Le Touzé (Eds)

# SHIP SCALE SELF PROPULSION CFD SIMULATION RESULTS COMPARED TO SEA TRIAL MEASUREMENTS

**Vuko Vukčević*, Hrvoje Jasak, Inno Gatin, Tessa Uroić**

University of Zagreb, Faculty of Mechanical Engineering and Naval Architecture,
Ivana Lučića 5, Zagreb, Croatia
e-mail: vuko.vukcevic@fsb.hr, hrvoje.jasak@fsb.hr, inno.gatin@fsb.hr, tessa.uroic@fsb.hr

**Key words:** CFD, Ship Scale Self Propulsion, Sea Trial Validation, OpenFOAM

**Abstract.** CFD simulation results for self propelled full scale ship are compared to sea trial measurements in this work. Two–phase RANS based CFD numerical model used in this work is based on the Ghost Fluid Method for numerically robust treatment of discontinuities at the free surface and the algebraic Volume–of–Fluid method for interface capturing. The propeller is modelled as a pressure–jump based actuator disc, allowing CPU time efficient simulations while preserving the accuracy of integral results. The numerical model is implemented in foam–extend, a community driven fork of the OpenFOAM software. The comparison with sea trials includes achieved forward speed, thrust and torque for given shaft speed (in RPM) for a general cargo carrier.

## 1 INTRODUCTION

As witnessed by many recent workshops [1, 2, 3], CFD is getting increasing attention in ship hydrodynamics. Traditionally, CFD is validated against experiments performed in model scale, mostly due to availability of data. The ability to accurately predict the self–propulsion point of the ship is of great importance in the marine industry, especially considering the current regulations regarding lower greenhouse gas emissions. Model scale CFD studies for a self–propelled ship still represent an active area of research (*e.g.* [4, 5, 6]), although it has been recently noted by Castro *et al.* [7] that the flow field at model and full scale is significantly different, especially in the stern area. According to Castro *et al.* [7], who performed both full and model scale CFD self–propulsion simulations with discretised propeller, the thinner boundary layer at full scale causes a more uniform inflow to the propeller with larger effective advance coefficient, thus increasing the propeller performance.

The availability of data from sea trials is scarce compared to experimental measurements, making a direct comparison of CFD with sea trials difficult. However, there is an ongoing effort for such a direct comparison, as discussed by Ponkratov and Zegos [8, 9], who validated their full scale CFD results against sea trials for a medium range tanker, obtaining good results. Recently, a Workshop on ship scale hydrodynamic computer simulation [10, 11] has been organised by the Lloyd's Register, providing a unique opportunity for all CFD practitioners to directly compare their simulation results with sea trial measurements.

The self–propelled ship in CFD can be modelled in a number of ways. The most efficient approach is to model the propeller as an actuator disc, which has been successfully used by many authors (see *e.g.* Tzabiras *et al.* [12]). On the other end, fully discretised, rotating propeller with either sliding interface [9] or dynamic overset grids [13, 14] presents the most detailed approach for simulating a self–propelled ship. Since such a detailed approach requires significant computational time, researchers have come up with different ways to speed up their computations without significantly affecting the results. For example, Ponkratov and Zegos [9] first ran their sliding interface CFD simulations with the Multiple Reference Frame (MRF) approach without rotating the propeller until the free surface stabilises, and then turned to full propeller rotation. Also very recently, Carrica *et al.* [13] presented a partially rotating frame approach, which allowed them to increase the time step with fully discretised propeller by one order of magnitude, while still being able to model a part of the propeller rotation. In this paper, we aim to primarily validate the achieved speed of the ship, where the CPU time efficiency is the most important factor and the detailed flow features behind the propeller are not of primary interest. For this reason, we employ the actuator disc model [15] for the full scale self–propulsion CFD simulations.

The paper is organized as follows. First, the mathematical and numerical model of the two–phase, RANS CFD code `navalFoam` is presented. The set–up of self–propulsion CFD simulations is presented next, where the results are directly compared to sea trial measurements. Finally, a short conclusion is given outlining the practicality and accuracy of such computations.

## 2 MATHEMATICAL AND NUMERICAL MODELLING

The CFD model is based on a single–equation formulation for the two–phases: water and air. Water and air are treated as incompressible, where the jump conditions at the free surface are taken into account with the Ghost Fluid Method (GFM). Following Huang *et al.* [16], GFM is used to discretise the jump conditions at the free surface, yielding interface–corrected discretisation schemes near the interface. For additional details regarding interface–corrected discretisation schemes in arbitrary polyhedral Finite Volume (FV) framework and the details on how they actually model an infinitesimally sharp free surface with respect to pressure and density discontinuity, the reader is referred to Vukčević [17]. The free surface is captured using the algebraic Volume–of–Fluid (VOF) method with interface compression [18], while the turbulence is modelled with a two–equation $k - \omega \; SST$ turbulence model.

The set of seven coupled, non–linear governing equations is discretised using arbitrary polyhedral FV support in foam–extend–4.0, a community driven fork of the open source software OpenFOAM. All time derivative terms are discretised with first order Euler scheme, while a second order linear upwind scheme is used for the convection term in the momentum equation. The face–centred values are obtained with central differencing with limited non–orthogonal correction for all diffusion terms. The non–linear coupling between equations is resolved in an iterative way using a combination of SIMPLE and PISO algorithms, where two SIMPLE steps encompass six PISO pressure corrections steps within a single time step. No effort was made in this paper to optimise the number of SIMPLE or PISO correctors, which will be the topic for future work.

## 3 SHIP SCALE SELF PROPULSION SIMULATIONS

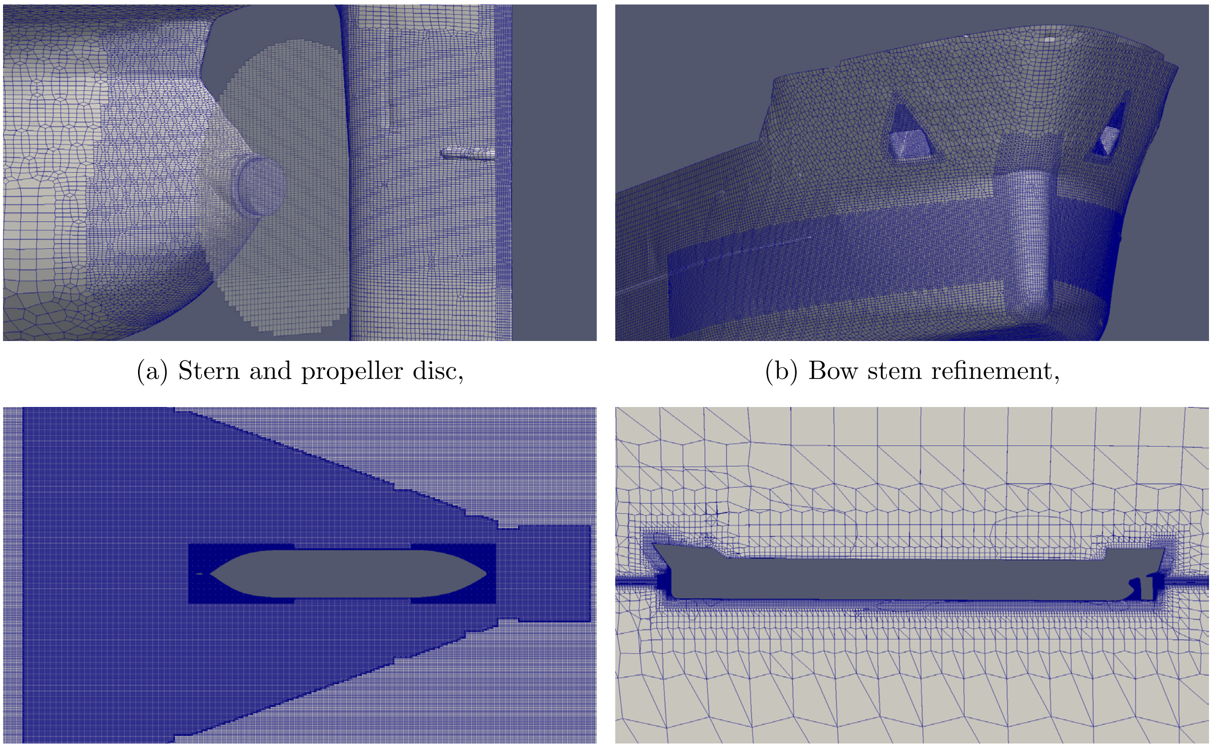
The full scale ship considered in this study is the REGAL general cargo carrier [11], where the particulars of the ship are summarised in Table 1. For the purposes of the Lloyd's Workshop on Ship Scale Hydrodynamic Computer Simulation [11], the ship has been dry–docked, its hull cleaned and propeller polished. The hull, rudder and propeller were 3D laser scanned in order to provide surface meshes to Workshop participants. Further details regarding the quality of hull and appendages can be found in [11]. After cleaning the hull and propeller, the ship was taken to sea trials in ballast condition, where three shaft speeds have been tested and the participants were requested to submit simulation results of achieved speed, thrust, torque, sinkage and trim. In this work, we present a set of results obtained with three grids for a single shaft speed (106.4 RPM) in order to assess numerical uncertainty.

Table 1: REGAL ship's particulars and trial conditions [11].

| | | |
|---|---|---|
| Length between perpendiculars | $L_{PP}$, m | 138 |
| Breadth moulded | $B$, m | 23 |
| Depth moulded | $D$, m | 12.1 |
| Propeller diameter | $D_P$, m | 5.2 (four bladed) |
| Service speed at design draught | $V$, kn | 14 |
| Water density | $\rho_w$, kg/m$^3$ | 1010 |
| Kinematic viscosity of water | $\nu_w$, m$^2$/s | $8.8394 \times 10^{-7}$ |
| Air density | $\rho_a$, kg/m$^3$ | 1.1649 |
| Kinematic viscosity of air | $\nu_a$, m$^2$/s | $1.6036 \times 10^{-5}$ |
| Longitudinal centre of gravity | $LCG$, m | 71.266 (from A.P.) |
| Vertical centre of gravity | $VCG$, m | 0.0 (free surface) |
| Transverse centre of gravity | $TCG$, m | -0.058 (starboard) |
| Mass | $\Delta$, t | 12881.27 |
| Pitch radius of gyration | $R_{yy}$, m | $0.25L_{PP}$ |

### 3.1 Computational grids

Three grids are generated with `cfMesh` [19], an open–source mesher which is available in foam–extend, a community driven fork of the open–source software OpenFOAM [20, 21]. Following Workshop instructions [11], the grids extend one $L_{PP}$ in front of the forward perpendicular, two $L_{PP}$ behind the aft perpendicular and one $L_{PP}$ from the starboard, portside and below the baseline. Compared to the real ship geometry, the superstructure and cranes are not modelled in order to keep the grid size to a minimum without significantly affecting the results. The propeller is modelled as a pressure jump–based actuator disc where the radial distribution of the pressure jump is modelled according to [15]. Detail of the fine grid at the stern is presented in Figure 1a, where one can see the circular grid interface of the actuator disc. Figure 1b presents bow stem refinement, while Figure 1c and Figure 1d present Kelvin angle refinement



(a) Stern and propeller disc, (b) Bow stem refinement, (c) Kelvin angle refinement, (d) Free surface and near hull refinement.

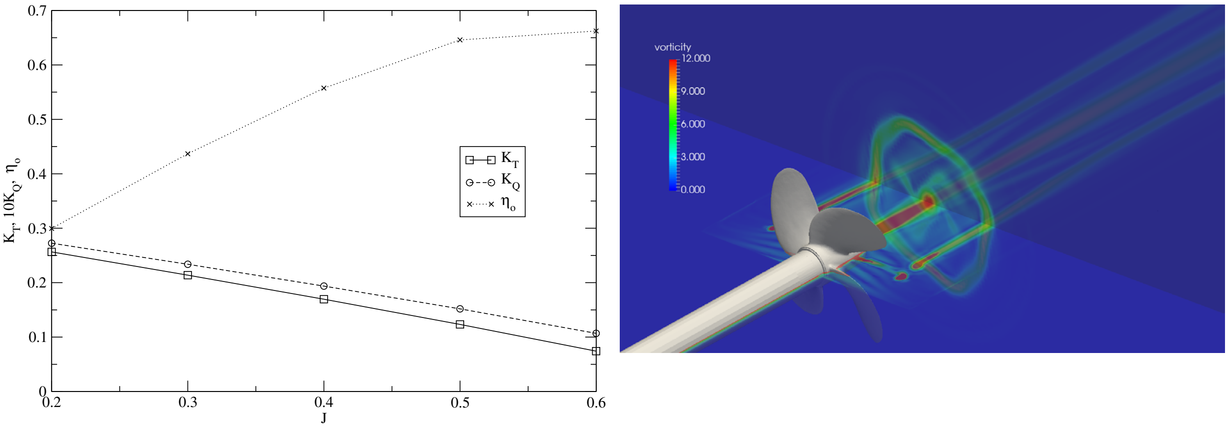
Figure 1: Details of the fine grid.

and aggressive refinement towards the free surface, respectively.

Three grids consist of approximately 5.6, 7.5 and 11.7 million cells, yielding a refinement ratio of $r_{mf} = 1.2$ between medium and fine grid and $r_{mc} = 1.1$ between coarse and medium grid. Approximately 95% of the cells are hexahedral, remaining 5% being general polyhedral cells with a small amount (0.03%) of prisms, pyramids and tetrahedra. Maximum non–orthogonality for the fine grid is 88.8°, while the average value is 7.3°. Six layers have been used for the boundary layer, with growth ratio of 1.3. Average dimensionless distance $y^+$ at the hull varies from 900 to 1100 between the three grids.

### 3.2 Open water propeller simulations

Full scale simulations of the propeller in open water are carried out to determine thrust and torque curves necessary for the actuator disc model. The two computational grids are created with `cfMesh`, where the GGI (Generalised Grid Interface) [22] is used to couple non–conformal grids at the rotating interface. The MRF approach is used to model the propeller rotation in a steady–state manner. Thrust and torque curves are obtained by performing five simulations for different advance ratios $J = V_a/(nD_p)$: 0.2, 0.3, 0.4, 0.5 and 0.6. The advance ratio is varied by changing the advance speed $V_a$, while keeping the propeller rotation rate $n = 71.62$ RPM constant. The open water results are presented in Figure 2a in terms of thrust




(a) Open water CFD results,

(b) Vorticity field generated by the propeller.

Figure 2: CFD results for the open water test.

coefficient $K_T = T/(\rho n^2 D_p^4)$, torque coefficient $K_Q = Q/(\rho n^2 D_p^5)$ and open water efficiency $\eta_o = JK_T/(2\pi K_Q)$. Figure 2b presents the vorticity field generated by the propeller for the $J = 0.4$ case.

### 3.3 Self–propulsion simulation results

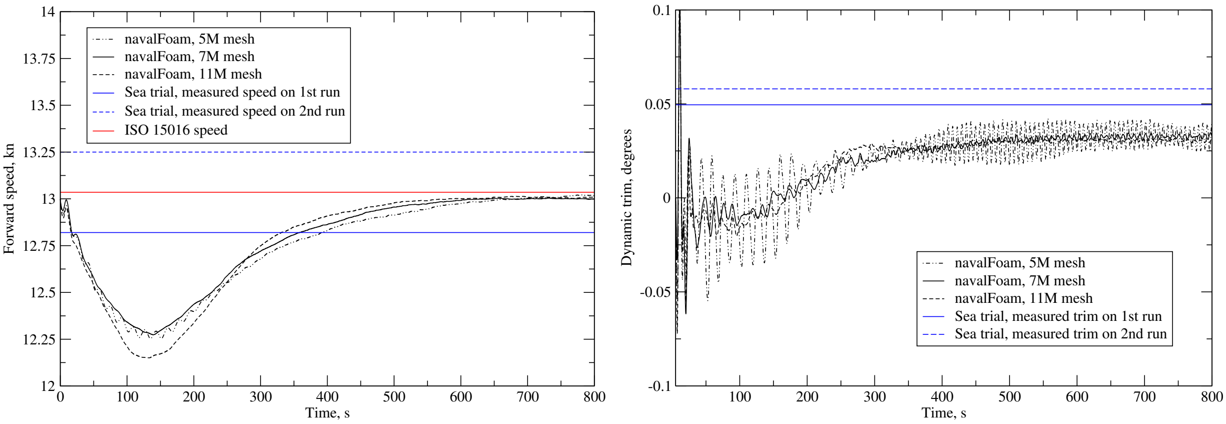
The approximate ship speed for the initial condition is 13 knots according to the Workshop's instructions [11], modelled as a free stream in this work. Hence, the CFD calculates the difference between prescribed 13 knots and the achieved speed. The ship is free to surge, heave and pitch, while sway, roll and yaw have been constrained. Since a steady state solution is sought, a fixed time step of $\Delta T = 0.075$ s has been used, yielding a maximum Courant–Friedrichs–Lewy number of $CFL = \mathcal{O}(10^2)$ during the simulation, where the mean $CFL$ number is of the order of $\mathcal{O}(10^{-1})$.

The converged solution on all three grids is achieved after 750 seconds, equivalent to 10 000 non–linear iterations. The convergence of forward speed of the ship is presented in Figure 3a, comparing the results with two sea trial runs. Results obtained with three grids fall within the range spanned by sea trials, where the difference between the ISO 15016 value and the CFD result on the fine grid is 0.2%. Moreover, since the oscillatory convergence is achieved with respect to grid refinement, the numerical uncertainty of the forward speed is calculated as:

$$U_v = 0.5 F_s \left( \max(V) - \min(V) \right) \, , \tag{1}$$

where $F_S = 3$ is the safety factor and $\max(V)$ and $\min(V)$ are the maximum and minimum speeds, respectively, achieved with the three grids. Using the results from Figure 3a, the corresponding numerical uncertainty is approximately 0.02 knots, or 0.15% of the fine grid result. Note that the final result oscillates at most by $\pm 0.0075$ knots, indicating the degree of iterative uncertainty for the measured ship speed.

The convergence of dynamic trim is presented in Figure 3b, where the calculated values under–predict the measured trim by 0.02 degrees for the first sea trial run, and 0.028 degrees




(a) Convergence of forward speed,

(b) Convergence of dynamic trim,

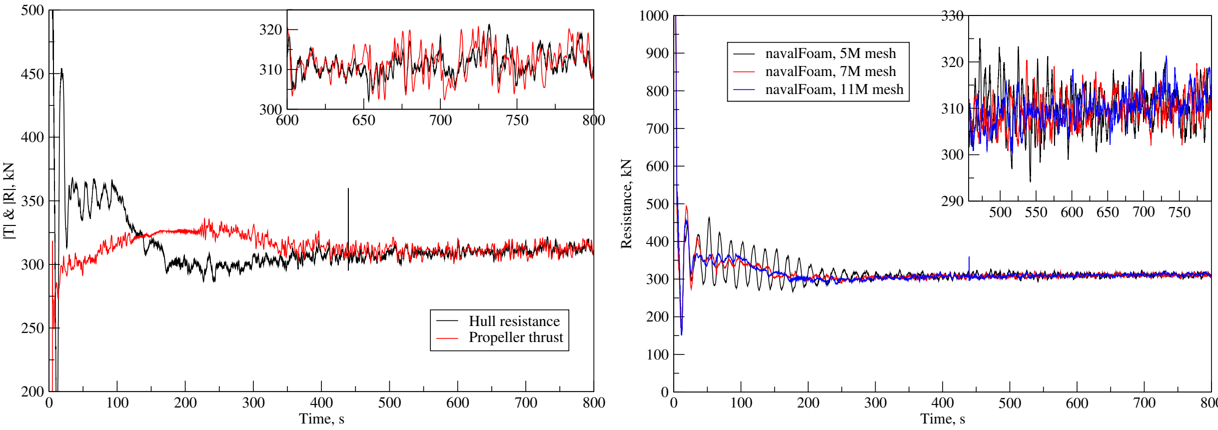
Figure 3: CFD results compared to sea trials.

for the second. Iterative uncertainty for dynamic trim, calculated with (1) reveals that the uncertainty is 0.025 degrees for the coarse grid, which is lowered to 0.0028 degrees using the fine grid.

The absolute values of resistance and propeller thrust obtained with the fine grid are presented in Figure 4a. Propeller thrust and resistance oscillate with $\pm 8$ kN, or $\pm 2.5\%$ compared to the solution averaged over past 200 hundred iterations. In the beginning of the simulation, the resistance is significantly larger than the propeller thrust, which causes the ship to slow down as indicated in Figure 3a. After approximately 2 and a half minutes, the propeller thrust starts to become larger than the resistance, thus causing the acceleration of the ship. Finally, after 10 minutes, propeller thrust and resistance are well balanced and the velocity of the ship does not change significantly (see Figure 3a).

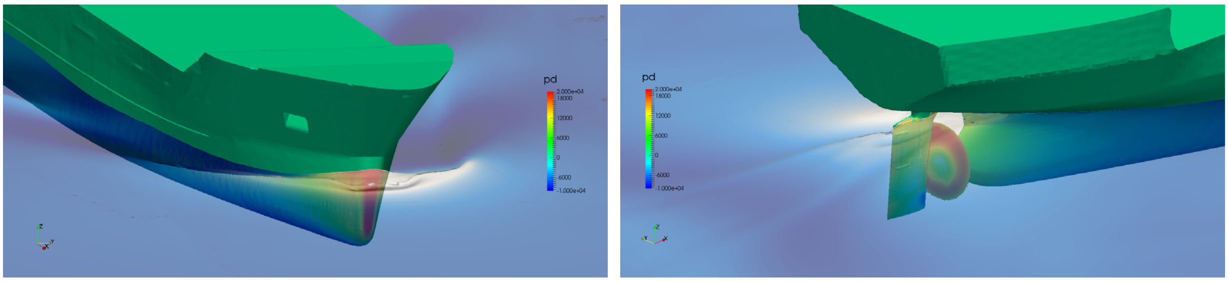
The convergence of resistance with coarse, medium and fine grids is presented in Figure 4b. The convergence is significantly less oscillatory on medium and fine grids compared to the coarse grid. The reason for such strongly oscillatory convergence on the coarse grid is the insufficient grid resolution near the free surface, producing small numerical waves. As can be seen from the zoomed view in Figure 4b, numerical results are insensitive to the mesh refinement, where the iterative uncertainty of approximately 6.5% is significantly larger than the grid uncertainty. Note that authors believe that the iterative uncertainty is caused by the innate unsteadiness of the flow at such a large length scales, rather than the iterative procedure for the non–linear equations sets. The viscous force obtained with the fine grid is approximately 137.071 kN, which is 2% below the ITTC 1957 correlation for the achieved speed. Figure 5a presents the dynamic pressure field at the bow, where a breaking bow wave can be seen due to vertical, cylindrical bow stem without a bulb. The dynamic pressure field at the stern and at the propeller plane represented by the actuator disc is shown in Figure 5b. Note that the dynamic pressure jump at the propeller plane is not symmetric due to the static roll angle as measured on sea trials [11], which is taken into account in present CFD simulations.

CPU times for all self propulsion simulations are summarised in Table 2. All simulations are carried out in parallel using up to 7 nodes (56 cores) on a distributed memory computational




(a) Convergence of resistance and thrust for fine grid, (b) Convergence of resistance for the three grids.

Figure 4: Convergence of forces in CFD simulations.



(a) Dynamic pressure at the bow, (b) Dynamic pressure at the stern.

Figure 5: Dynamic pressure field in bow and stern regions.

cluster: CPU–2x Intel Xeon E5-2637 v3 4–core, 3.5GHz, 15MB L3 Cache, DDR4–2133, with InfiniBand communication. Convergence is achieved after 39.4 hours (1.6 days) wall clock time for the coarse grid and after 83.8 hours (3.5 days) for the fine grid. It is interesting to note that one second of real time for such a full scale simulation requires $\mathcal{O}(10^2)$ seconds of CPU time using these computational resources. The main enabler for a large time step of $\Delta t = 0.075$ s is the actuator disc model. If the discretised, rotating propeller is used, the time–step would be approximately 100 times lower in order to resolve propeller motion with 0.5 degrees per time step (*e.g.* as used by Shen *et al.* [14]).

## 4 CONCLUSION

This paper presents a direct comparison of ship scale CFD simulations with sea trial measurements. A single case is simulated where the ship is free to sail given a constant propeller revolution rate. The CFD simulations are performed with an actuator disc model, for which the thrust and torque curves are generated with ship scale open water simulations using the MRF approach. Three computational grids are used, ranging from 5 to 11 million of cells in order to

Table 2: CPU times for self propulsion simulations.

| Grid | Coarse | Medium | Fine |
|---|---|---|---|
| Number of cells | 5 597 931 | 7 469 642 | 11 727 781 |
| Number of cores | 48 | 48 | 56 |
| CPU time per time–step, s | 14.2 | 19.1 | 29.8 |
| CPU time per second of real time | 189.8 | 254.5 | 397.2 |
| CPU time until convergence ($t = 750$ s), h | 39.4 | 53.1 | 83.8 |

assess numerical uncertainty.

CFD results for the achieved ship speed and dynamic trim compare well with two sea trial measurements, where the grid refinement study reveals low numerical uncertainties. The discrepancy between successive CFD simulations using three grids is smaller than the difference between two sea trial measurements, which is expected since the weather and other conditions cannot be fully controlled during a sea trial. The successful comparison with sea trials indicates that the present CFD model with actuator disc for propeller can be readily used to estimate the ship speed, thus avoiding the need of CPU–time consuming discretised, rotating propeller.

Although the overall CPU time of approximately 7 days for three simulations can be considered reasonable for ship scale simulations, the future effort shall be focused on investigating the possibility of using smaller and coarser grids with more suitable refinement regions in order to further decrease the CPU time.

## 5 ACKNOWLEDGMENTS

Considering how difficult it is to find publicly available results from sea trials, we would like to thank and acknowledge Lloyd's Register for organising and hosting the Workshop on ship scale hydrodynamic computer simulation [10] and making the valuable data from sea trials available, thus enabling us to perform a direct comparison of CFD with sea trials.

## REFERENCES

[1] L. Larsson, F. Stern, M. Visonneau, N. Hirata, T. Hino, J. Kim (Eds.), Tokyo 2015: A Workshop on CFD in Ship Hydrodynamics, Vol. 2, NMRI (National Maritime Research Institute), Tokyo, Japan, 2015.

[2] L. Larsson, F. Stern, M. Visonneau, Numerical Ship Hydrodynamics: An assessment of the Gothenburg 2010 workshop, Springer, 2013. `doi:10.1007/978-94-007-7189-5`.

[3] SIMMAN 2014: Workshop on Verification and Validation of Ship Manoeuvring Simulation Methods, `http://www.simman2014.dk`, [Online; accessed 26 September 2014] (2014).

[4] P. Carrica, A. Castro, F. Stern, Self–propulsion computations using a speed controller and a discretized propeller with dynamic overset grids, J. Mar. Sci. Techno. 15 (2010) 316–330. `doi:10.1007/s00773-010-0098-6`.

[5] Y. Xing-Kaeding, S. Gatchell, Resistance and Self–Propulsion Predictions for Japan Bulk Carrier without and with Duct using the FreSCo+ code, in: Proceedings of the Tokyo 2015: A Workshop on CFD in Ship Hydrodynamics, Vol. 3, 2015, pp. 291–296.

[6] G.-H. Kim, J.-H. Jun, Numerical Simulations for Predicting Resistance and Self–Propulsion Performances of JBC using OpenFOAM and Star-CCM+, in: Proceedings of the Tokyo 2015: A Workshop on CFD in Ship Hydrodynamics, Vol. 3, 2015, pp. 285–296.

[7] A. Castro, P. M. Carrica, F. Stern, Full scale self-propulsion computations using discretized propeller for the KRISO container ship KCS, Comput. Fluids 51 (2011) 35–47. doi:10.1016/j.compfluid.2011.07.005.

[8] D. Ponkratov, C. Zegos, Ship Scale CFD Self–Propulsion Simulation and Its Direct Comparison with Sea Trial Results, in: Proceedings of the International Conference on Computational and Experimental Marine Hydrodynamics (MARHY'14), 2014.

[9] D. Ponkratov, C. Zegos, Validation of Ship Scale CFD Self–Propulsion Simulation by the Direct Comparison with Sea Trial Results, in: Proceedings of the Fourth International Symposium on Marine Propulsors, 2015.

[10] Lloyd's Register: A workshop on ship scale hydrodynamic computer simulation, http://www.lr.org/en/news-and-insight/events/ship-scale-hydrodynamics-numerical-methods-workshop.aspx, [Online; accessed 22 February 2017] (2016).

[11] D. Ponkratov (Ed.), Proceedings: 2016 Workshop on Ship Scale Hydrodynamic Computer Simulations, Lloyd's Register, Southampton, United Kingdom, 2017, http://www.lr.org/en/projects/findings-of-lrs-full-scale-numerical-modelling-workshop.aspx [Online; accessed 24 February 2017].

[12] G. Tzabiras, S. Polyzos, G. Zarafonitis, Self–Propulsion Simulations of Passenger–Ferry Ships with Bow and Stern Propulsors, in: Proceedings of the 12th Numerical Towing Tank Symposium (NUTTS), 2009.

[13] P. M. Carrica, A. Mofidi, E. Martin, Progress Toward Dircet CFD simulation of Manoeuvres in Waves, in: Proceedings of the MARINE 2015 Conference, 2015, pp. 327–338.

[14] Z. Shen, D. Wan, P. M. Carrica, Dynamic overset grids in OpenFOAM with application to KCS self–propulsion and maneuvering, Ocean Eng. 108 (2015) 287–306. doi:10.1016/j.oceaneng.2015.07.035.

[15] E. Svenning, Implementation of an actuator disk in OpenFOAM, Tech. rep., Chalmers Universtity of Technology (July 2010).

[16] J. Huang, P. M. Carrica, F. Stern, Coupled ghost fluid/two–phase level set method for curvilinear body–fitted grids, Int. J. Numer. Meth. Fluids 44 (2007) 867–897. doi:10.1002/fld.1499.

[17] V. Vukčević, Numerical modelling of coupled potential and viscous flow for marine applications - in preparation, Ph.D. thesis, Faculty of Mechanical Engineering and Naval Architecture, University of Zagreb, PhD Thesis (2016). doi:10.13140/RG.2.2.23080.57605.

[18] H. Jasak, V. Vukčević, D. Christ, Rapid Free Surface Simulation for Steady-State Hull Resistance with FVM using OpenFOAM, in: Proceedings of the 30th Symposium on Naval Hydrodynamics, 2014, pp. 548–554.

[19] F. Juretić, cfMesh: Advanced Meshing Tool, cfMesh.com, [Online; accessed 22 February 2017] (2017).

[20] H. G. Weller, G. Tabor, H. Jasak, C. Fureby, A tensorial approach to computational continuum mechanics using object oriented techniques, Comput. Phys. 12 (1998) 620–631.

[21] H. Jasak, A. Jemcov, Z. Tuković, OpenFOAM: A C++ library for complex physics simulations, International Workshop on Coupled Methods in Numerical Dynamics, IUC, Dubrovnik, Croatia.

[22] M. Beaudoin, H. Jasak, Development of Generalized Grid Interface for Turbomachinery Simulations with OpenFOAM, in: Proceedings of the Open Source CFD International Conference, 2008.